

**KEMI-TORNIO UNIVERSITY OF APPLIED SCIENCES
TECHNOLOGY**

Yu Fan

Cascading Style Sheets

The items that should remain confidential and the length of time of confidentiality are mentioned here.

The Bachelor's Thesis Technology programme
Kemi 2010

PREFACE

I want to thank my supervisor Aalto Teppo who gave me great help and good advice on my thesis modification.

ABSTRACT

Kemi-Tornio University of Applied Sciences, Technology	
Degree Programme	Information technology Engineering
Name	Yu Fan
Title	Cascading Style Sheets
Type of Study	Bachelor's Thesis
Date	11 November 2010
Pages	26 + 3 appendices or appendixes
Instructor	Aalto Teppo

The topic of this thesis is style sheet languages for structured documents on the web. Style sheet languages that were developed and used prior to the web are analyzed and compared with style sheet proposals for the web between 1993 and 1996. The thesis describes the design of a web-centric style sheet language known as Cascading Style Sheets (CSS). CSS has several notable features including: cascading, pseudo-classes and pseudo-elements, forward-compatible parsing rules, support for different media types, and a strong emphasis on selectors. Problems in CSS are analyzed, and recommended future research is described.

Due to characteristics of the web including a screen-centric publishing model, a multitude of output devices, uncertain delivery, strong user preferences, and the possibility for later binding between content and style – the hypothesis is that the web calls for different style sheet languages than does traditional electronic publishing.

The result of this work is to describe style sheet languages for structured documents on the web. The hypothesis is that the web calls for different style sheet languages than does traditional electronic publishing. Further, the design of a style sheet language that fulfils the specific requirements of the web

Key words: Cascading Style Sheets, HTML, Style sheet languages and structured documents.

TABLE OF CONTENTS

1. Introduction	1
1.1 Back ground	1
1.2 Overview of the thesis.....	1
2.1 Structure and presentation.....	3
2.1.1 Abstraction levels.....	3
2.1.2 Presentational HTML.....	3
2.2 Style sheets.....	4
2.2.1 Style sheet mechanisms for the web	4
2.3 Cascading Style Sheets (CSS).....	4
2.3.1 The CSS development.....	5
3. Structured documents.....	6
3.1 Abstraction levels.....	6
3.2 Structured document systems.....	7
3.3 Hyper Text Markup Language (HTML)	8
3.4 HTML in context.....	8
4. Cascading Style Sheets(CSS).....	9
4.1 The advantages of CSS	9
4.2 Syntax.....	9
4.3 Selector.....	11
4.3.1 Type selectors.....	11
4.3.2 ID selectors.....	11
4.3.3 Child selectors	11
4.3.4 Universal selector.....	12
4.3.5 Class selectors	12
4.3.6 TAG Selector	12
4.4 Properties	12
4.4.1 Font	12
4.4.2 Color.....	13
4.4.3 Background	13
4.4.4 Width.....	13
4.4.5 Height.....	13
4.4.6 Float	14
4.4.7 Clear	14
4.4.8 Display	14
4.5 Length Units.....	14
4.6 URLs	15
5. Problems in CSS and Internet explorer.....	16
5.1 Ghost Text bug.....	16
5.2 Position Relative and Overflow Hidden	16
5.3 Errors in the specifications.....	17
5.4 Cascading problems	17
5.5 User interface problems	18
5.6 Complexity problems	18
5.7 Problems in implementations.....	18
5.8 Bicubic Image Scaling	18

6. Mobile CSS	20
6.1 Usage and browser support	20
6.2 CSS support in CSS Mobile Profile 2.0	21
7. Conclusions	23
8. References	25
9. List of appendices	27

1. INTRODUCTION

1.1 Back ground

CSS is a new "old technology." In the Internet field, any kind of technology which was born 10 years ago can be regarded as "old technology", but CSS is a special case, it is a new and old technology, because in almost 10 years after the its birth, CSS makes people truly realize its value.

There are two kinds of motivations on studying and researching a technology, one is working, and the other is interest. CSS does not only can be used to work for a life, more worthy of love with this technology, because it offers unlimited creative space.

The topic of this thesis is style sheet languages for structured documents on the web. The hypothesis is that the web calls for different style sheet languages than does traditional electronic publishing. Further, the design of a style sheet language that fulfils the specific requirements of the web, namely Cascading Style Sheets is described. In my thesis, I will explain some basic concepts but they often have misunderstood, and discuss how to make HTML and CSS to keep clear and good structured.

1.2 Overview of the thesis

Introduction of HTML and CSS

This chapter is an introduction to the topic of the thesis and related subjects. The historical context in which CSS was developed is described, including the development of HTML.

Structured documents

Style sheet languages and structured documents are mutually dependent. Without style sheets, structured documents cannot be presented and without structured documents there is nothing for style sheets to present.

Cascading Style Sheets

This chapter marks the start of the "how" section of the thesis. In this chapter, Cascading Style Sheets (CSS) is described in some details.

Problems in CSS

This chapter discusses problems in, and related to, the CSS specifications. These range from simple spelling errors to more complex questions such as whether or not some functionality fulfils its intended role.

Mobile CSS

Most web pages are written for desktop computers with large colour monitors. This chapter describes how cascading can be used to render web pages on small screens.

Conclusions

The conclusions support the argument of the thesis: due to its characteristics, the web calls for style sheet languages different from those for traditional electronic publishing.

2. HTML AND CSS

HTML (Hyper Text Mark-up Language) or Hyper Text Markup Language, is the most widely used web language, but also constitutes the main language of Web documents. HTML command text is composed of descriptive text, HTML commands that: text, graphics, animation, sound, tables, and links. HTMLS' structure consists of the head (Head), body (Body) two parts, the head which the information needed to describe the browser, while the body contains the specific content to be described. /1/

2.1 Structure and presentation

Scientific environments value logic, structure and content more highly than aesthetics, imagery and style. This sense of structure is reflected in HTML. Each paragraph is marked as such and headings are given a numbered level to indicate their place in the document structure. /2/

As the web attracted attention outside of scientific environments, authors started complaining that they did not have enough influence over the appearance of their pages. One of the most frequent questions asked by authors new to the web was how to change fonts and colors of elements. /2/

2.1.1 Abstraction levels

By adding presentational tags to HTML, the language evolved from being an abstract, structured, mark up language where authors marked the different logical roles of the text (paragraphs, headlines, lists and so forth) towards a concrete presentation language where emphasis is on the final form presentation of documents (fonts, colours and layout). /3/

In traditional paper-based publishing, the reader receives a final form product. Each letter on a printed page has a fixed position, shape, size and color that cannot be changed by the reader. Electronic documents, however, are unfinished products that must be assembled before they can be presented to the human reader. In the assembly process – better known as formatting many choices of how to present the document are made. For example, the browser must pick the fonts and colors to use when presenting the document on a color screen. The level of processing that an electronic document needs will vary considerably depending on what document format is used. As such, electronic documents are similar to furniture: some furniture comes pre-assembled while other items are bought in flat packages and the owner must do the final assembly. If a document format requires much processing, it is said to have a high level of abstraction. If the document format needs little processing, it is said to have a low level of abstraction. /3/

2.1.2 Presentational HTML

The introduction of presentational tags in HTML was a downwards move on the ladder

of abstraction. Several of the new elements were meaningful only for particular output devices. The creators of HTML intended it to be usable in many settings but presentational tags threatened device independence, accessibility and content reuse. /3/

The development of HTML into a presentation-oriented language also changed the power balance between authors and users. Structured documents must be formatted by the browser before presentation, and – to some extent – the formatting process can be influenced by the user. However, when the browser receives a document in its final form, the formatting process is complete and can no longer be influenced by the user. /3/

2.2 Style sheets

Style sheets represent a major breakthrough for Web page designers, expanding their ability to improve the appearance of their pages. Style sheets describe how documents are presented on screens, in print, or perhaps how they are pronounced. W3C has actively promoted the use of style sheets on the Web since the Consortium was founded in 1994. The Style Activity has produced several W3C Recommendations (CSS1, CSS2, XPath, and XSLT). CSS especially is widely implemented in browsers. These techniques work for some of the people, some of the time, but not for all of the people, all of the time. They include:

- Using proprietary HTML extensions
- Converting text into images
- Using images for white space control
- Use of tables for page layout
- Writing a program instead of using HTML /3/

2.2.1 Style sheet mechanisms for the web

A crude form of style sheets was hard-coded into the first WWW client implemented on the NeXT machine at CERN. However, no specification for style sheets was written and no syntax for a style sheet language was proposed; it was considered a matter for each browser to decide how to best display pages to its users. /4/

The potential benefits of using style sheets on the web are significant. A well-developed style sheet mechanism would give authors a richer stylistic vocabulary than they could hope for in an evolving HTML. Also, HTML would remain a structured mark up language that worked on a wide range of devices. /4/

2.3 Cascading Style Sheets (CSS)

Cascading Style Sheets (CSS) is a style sheet technology. When CSS is used in the home page design, it can achieve a more precise control on the page layout, fonts, colors,

backgrounds and other effects. To do some simple modifications on the code, can change the different parts of the same page or different pages' layout and format. /5/

Its role can be achieved:

1. Can be used in almost all browsers.
2. Can faster download pages.
3. Make the fonts more beautiful in the webpage, make the page really enjoyable.
4. Can easily control the layout of the page.
5. Modify the CSS file in the appropriate line, and then all the pages throughout the site will follow the change. /5/

2.3.1 The CSS development

From the early 1990s, the invention of HTML style sheets began to appear in various forms, and different browsers with their own style sheet language, the user can use these style sheet languages to regulate the displayed mode. In the Beginning, style sheet is designed for the reader; the first HTML version contains only a few display properties, the reader decide how the page should be displayed. But with the growth of HTML, in order to meet the requirements of designers, HTML received a lot of display functions. /6/

There are various versions of CSS (or Level), so it's very important to know which version should be used. In late 1996, CSS 1 became the recommendation standard, which contains very basic attributes, such as fonts, colors, blank edge. On this basis, CSS2 adds advanced concepts (such as floating and positioning) and the advanced selectors (such as sub selectors, adjacent selectors and common selectors). CSS2 is still the latest version of CSS even it has been a recommended standard in 1998. World Wide Web Consortium (W3C) has very slowly to development CSS3 in 2000, but from the final release still have a long time to go, to enhance the development and implementation of the speed for browser, CSS3 divided into modules, these modules can be independent publication and implementation. CSS3 contains some exciting new features, including a module for multi-column layout. But it still not release at the moment. The newest version of CSS is CSS 2.1; it more accurately reflects the current state of CSS. /6/

3. STRUCTURED DOCUMENTS

Style sheet languages and structured document formats are mutually dependent on each other. Without style sheets, structured documents cannot be presented and without structured documents there is nothing for style sheets to present. Due to the strong relationship between the two, it is important to understand structured documents when studying style sheet languages. Some structured document systems that have been most influential on style sheet languages are discussed in this chapter. /7/

3.1 Abstraction levels

A similar example of abstraction levels can be found in the field of computer networking. In 1983, the International Standards Organization (ISO) developed a network model called Open Systems Interconnection (OSI) Reference Model which defined a framework of computer communications. The ISO/OSI Reference Model has seven layers, each of which has a different level of abstraction. The seven layers are: physical, data link, network, transport, session, presentation and application. /8/

The notion of an abstraction ladder is useful when evaluating document formats. How high a certain document format is on the ladder will determine the complexity of formatting the document into a presentation. Since the formatting of a document is specified by a style sheet, the abstraction level is a crucial feature for the success of style sheets. /8/

The vertical nature of a ladder corresponds to how one describes abstraction levels as high or low. Typical characteristics of document formats that are high on the ladder of abstraction are:

- The information needs processing in order to be presented. For example, in order to render an HTML document visually, the words must be broken into lines, fonts must be selected, and the characters must be turned into rasterized glyphs.
- The information can be processed and presented in many different ways. Presenting a document visually is only one of several possibilities; others include aural renderings and Braille embedding.
- The information is represented in a compact manner. Representing a character with an eight-bit code is more compact than representing an image of the same character. /8/

3.2 Structured document systems

Beginning around 1980, there was an active research community in the field of electronic publishing and structured documents. The community published their results in the proceedings of the Electronic Publishing conferences, in the journal

	GIF, PNG	private XML vocabulary	PDF	XSL-FO	HTML	MathML
application-specific semantics?	no	no	no	no	no	yes
device-independent?	no	no	no	no	yes	yes
roles known?	no	no	no	no	yes	yes
text in logical order?	unknown	unknown	no	yes	yes	yes
reflow possible?	no	unknown	no	yes	yes	yes
scalable?	no	unknown	yes	yes	yes	yes
text machine-readable?	no	yes	yes	yes	yes	yes
text human-readable?	yes	yes	yes	yes	yes	yes

Figure 1: A comparison of document formats on the ladder of abstraction. /9/

The researchers generally agreed on the benefits of vendor-neutral document formats to facilitate document exchange. The benefits of structured documents were also well understood. There were, however, several approaches to structured documents, and competing formats were developed. /9/

Figure 1 shows the relative positions of various document formats on the ladder of abstraction. Here are some notes to the figure:

- GIF [GIF 1990] and PNG [PNG 1996] are bitmap image formats rather than document formats, but images are often used to represent documents. Fax transmission is a common example outside the web.
- PDF [Adobe 1993] is a document format developed by Adobe Systems. PDF is a presentation-oriented format and has no concept of, for example, paragraphs and headings. Many users have discovered this when trying to copy content from PDF documents laid out in several columns. When selecting text, the selection will span across multiple columns and thereby mix text from several parts of the document into the same selection. Recent versions of PDF have introduced functionality to retain a document's logical structure in PDF [Adobe 2001].

- XSL-FO refers to a document consisting of formatting objects as defined in the XSL Recommendation [XSL 2001].
- XML [XML 1998], in which several of the emerging formats are written, is also included in the table and refers to documents published using private XML vocabularies where the semantics are not universally known.
- The rating of HTML is based on a best-case scenario where the author makes use of semantic elements and does not alter the reading order of elements by using features such as positioning or tables. It may be argued that most HTML documents do not follow these conventions.
- MathML is a W3C Recommendation for mathematical notation [MathML 1998]. /9/

3.3 Hyper Text Markup Language (HTML)

The specification of HTML is along with HTTP and URL specifications, it's one of the basic components of the web. HTML has had large impact on how content is compiled, stored, transmitted and processed. The design of HTML is probably one of the main reasons for the success of the web. /10/

3.4 HTML in context

HTML has large development space from the first version made in 1991. Along the way, many functions have been added while ensuring backwards compatibility. The principle of encouraging logical, rather than presentational mark up has remained despite resistance from implementers' and authors. As a consequence, style sheets became necessary and later found their place on the web. /10/

4. CASCADING STYLE SHEETS(CSS)

CSS language is a markup language, it's not need to compile, and it can be directly executed by the browser. In the standard website design, CSS is responsible for web content performance. CSS file is a text file that contains some CSS tags, CSS files must use CSS for the file name suffix. To do some simple changes in CSS file can change the overall performance of web pages; thus reduce the workload. CSS is develop and maintenance by W3C's CSS working group. /11/

4.1 The advantages of CSS

In the CSS standard, not only has redefined the style of the original HTML, such as text size, colour, etc.; also joined the duplication of text, block such changes and be placed anywhere on a number of new properties. CSS can be used by richer, more flexible style, easier to design attractive web pages. It will also allow web page design and maintenance more efficient. /12/

To reduce the use of image files: Many web pages for the sake of design effect, while extensive to use of images, resulting in slow page download speed. CSS provides a lot of text style settings, and coupled with IE built-in filter effects, so you can easily replace the original image to the performance of visual effects. This design approach to modify the web content has become more convenient, but also greatly increased the download speed; /12/

Centralized management style information: The basic concept is that it can contents of a web page to show the performance of its style settings separately, that is, the appearance of a web page set up information from the page content independent, and centralized management. In this way, when to change the look of the page, simply change the style settings part, HTML document itself does not require any changes;

Share Style Settings: Web page style and content of the separation benefits set up, in addition to centralized management, the CSS style if further information is stored into a separate file, but also allow multiple page documents together to use it. /12/

Style categories will be used: as opposed to more than HTML documents can be applied with a CSS style file can also be in an HTML web page document to apply multiple CSS style file. /12/

4.2 Syntax

A Cascading Style Sheet consists of one or more statements conforming to CSS syntax. CSS is defined by three components: Selector attributes (properties) and values of the property (value). CSS uses a simple syntax. Here is an example:

```
H1 {font-size: 2em}
```

(In web optimization, often used H1 tags to optimize the keywords.)

The rule in the example above consists of two main parts: Tags (H1) and declaration (font-size: 2em). The declaration has two parts: a property (font-size) and a value (2em). While the example above tries to influence only one of the properties needed for rendering a document, it qualifies as a style sheet on its own. Combined with other style sheets it will determine the final presentation of the document. /13/

Several declarations can be grouped in a declaration block:

```
BODY {  
    margin: 3em;  
    font-family: "Gill Sans", sans-serif;  
}
```

Declarations inside the declaration block are separated by semicolons. The last declaration is optionally followed by a semicolon. The first declaration in the example above sets the margin around the BODY elements to be 3em. The em unit refers to the font size of the element. In this case, the result is that the margins around the BODY element are three times wider than the font size of the BODY element. The margin property is an example of a shorthand property which sets values on several other individual properties at the same time. /13/

The second declaration in the above example has a comma-separated list of font families as value. If the first value cannot be used, the next value will be tried, and so forth. Only some CSS properties accept lists as values. /13/

Selectors can also be grouped in comma-separated lists:

```
H1, H2 {  
    font-weight: bold;  
}
```

In the example above, the declaration block applies to both H1 and H2 elements. Most of the logic in CSS is expressed in selectors. Here is a more ambitious example:

```
DIV.ingress P: first-line {  
    text-transform: uppercase;  
}
```

In plain text, the rule above reads: the first line of all P elements inside DIV elements of class ingress should be transformed to uppercase. /13/

4.3 Selector

A simple selector is a type selector or universal selector followed immediately by zero or more attribute selectors, ID selectors, or pseudo-classes, in any order. The simple selector matches if all of its components match. A selector is a chain of one or more simple selectors separated by combinatory. Combinatory are: white space, ">", and "+". White space may appear between a combinatory and the simple selectors around it. /14/

4.3.1 Type selectors

A type selector matches the name of a document language element type. A type selector matches every instance of the element type in the document tree. A type selector containing a namespace prefix that has not been previously declared for name spaced selectors is an invalid selector.

e.g.:

```
h1 {font-family: sans-serif} /14/
```

4.3.2 ID selectors

ID selectors are similar to class selectors. They can be used to select any HTML element that has an ID attribute, regardless of their position in the document tree. Examples of ID selectors:

e.g.:

```
#content  
{  
font - size: 14px;  
line - height: 120%;  
} /14/
```

4.3.3 Child selectors

A child selector is used to select an element that is a direct child of another element. Child selectors will not select all descendants, only direct children. Child selector matches when an element is the child of some element. Child selector is made up of two or more selectors separated by ">".

The following rule sets the style of all P elements that are children of BODY:

```
body > P {line-height: 1.3}
```

The following example combines descendant selectors and child selectors:

```
div ol>li p /14/
```


4.3.4 Universal selector

The universal selector, written "*", matches the name of any element type. It matches any single element in the document tree. If the universal selector is not the only component of a simple selector, the "*" may be omitted. For example:

- *[lang=fr] and [lang=fr] are equivalent.
- *.warning and .warning are equivalent.
- #myid and #myid are equivalent. /14/

4.3.5 Class selectors

Compared with id, class can be reused, the same style for multiple elements, can be directly defined as a class. Class selectors can be used to select any HTML element that has a class attribute, regardless of their position in the document tree. /14/

XHTML:

```
<p class="he"></p>
<span class="he"></span>
<h5 class="he"></h5>
```

CSS:

```
.he
{
margin:10px;
background-color:red; /14/
```

4.3.6 TAG Selector

Tag selector is another simple method of CSS rules implementation. The tag selector displays the entire chain of tags that apply to the current selection or to the cursor position.

For example:

```
p {color:#999; font-weight:bold;} /15/
```

4.4 Properties

4.4.1 Font

The font property may be used as shorthand for the various font properties, as well as the line height. CSS font properties define the font family, boldness, size, and the style of a text. For example:

P {font: italic bold 12pt/14pt Times, serif} /16/

4.4.2 Color

CSS colors are defined using a hexadecimal (hex) notation for the combination of Red, Green, and Blue color values (RGB). The lowest value that can be given to one of the light sources is 0 (hex 00). The highest value is 255 (hex FF). The color property allows authors to specify the color of an element. See the Units section for color value descriptions. Some example color rules include:

```
H1 {color: blue}
H2 {color: #000080}
H3 {color: #0c0} /17/
```

4.4.3 Background

The background property is shorthand for the more specific background-related properties. CSS background properties are used to define the background effects of an element. Some examples of background declarations follow:

```
BODY    {background: white url(http://www.htmlhelp.com/foo.gif) }
BLOCKQUOTE { background: #7fffd4 }
P       { background: url(../backgrounds/pawn.png) #f0f8ff fixed }
TABLE   { background: #0c0 url(leaves.jpg) no-repeat bottom right } /18/
```

4.4.4 Width

Each block-level or replaced element can be given a width, specified as a length, a percentage, or as auto. (A *replaced element* is one for which only the intrinsic dimensions are known; HTML replaced elements include IMG, INPUT, TEXTAREA, SELECT, and OBJECT.) The initial value for the width property is auto, which results in the element's intrinsic width (*i.e.*, the width of the element itself, for example the width of an image). Percentages refer to the parent element's width. Negative values are not allowed.

This property could be used to give common widths to some INPUT elements, such as submit and reset buttons:

```
INPUT.button {width: 10em} /19/
```

4.4.5 Height

Each block-level or replaced element can be given a height, specified as a length or as auto. (A replaced element is one for which only the intrinsic dimensions are known; HTML replaced elements include IMG, INPUT, TEXTAREA, SELECT, and OBJECT.)

The initial value for the height property is auto, which results in the element's intrinsic height (*i.e.*, the height of the element itself, for example the height of an image). Negative lengths are not allowed.

As with the width property, height can be used to scale an image:

```
IMG.foo { width: 40px; height: 40px } /20/
```

4.4.6 Float

The float property allows authors to wrap text around an element. This is identical in purpose to HTML 3.2's ALIGN=left and ALIGN=right for the IMG element, but CSS1 allows all elements to "float," not just the images and tables that HTML 3.2 allows. /21/

4.4.7 Clear

The clear property specifies if an element allows floating elements to its sides. A value of left moves the element below any floating element on its left; right acts similarly for floating elements on the right. Other values are none, which is the initial value, and both, which moves the element below floating elements on both of its sides. This property is similar in function to HTML 3.2's <BR CLEAR=left|right|all|none>, but it can be applied to all elements. /22/

4.4.8 Display

The display property specifies if/how an element is displayed, and the visibility property specifies if an element should be visible or hidden. The display property is used to define an element with one of four values:

block (a line break before and after the element)

inline (no line break before and after the element)

list-item (same as block except a list-item marker is added)

none (no display) /23/

4.5 Length Units

A length value is formed by an optional + or -, followed by a number, followed by a two-letter abbreviation that indicates the unit. There are no spaces in a length value; e.g.,

1.3 em is not a valid length value, but 1.3em is valid. A length of 0 does not require the two-letter unit identifier. /24/

Both relative and absolute length units are supported in CSS1. Relative units give a length relative to another length property, and are preferred since they will better adjust to different media. The following relative units are available:

em (ems, the height of the element's font)
ex (x-height, the height of the letter "x")
px (pixels, relative to the canvas resolution)

Absolute length units are highly dependent on the output medium, and so are less useful than relative units. The following absolute units are available:

in (inches; 1in=2.54cm)
cm (centimeters; 1cm=10mm)
mm (millimeters)
pt (points; 1pt=1/72in)
pc (picas; 1pc=12pt) /24/

4.6 URLs

A URL value is given by url(foo), where foo is the URL. The URL may be optionally quoted with either single (') or double (") quotes and may contain whitespace before or after the (optionally quoted) URL. /25/

Parentheses, commas, spaces, single quotes, or double quotes in the URL must be escaped with a backslash. Partial URLs are interpreted relative to the style sheet source, not to the HTML source. /25/

Examples:

```
BODY { background: url(stripe.gif) }  
BODY { background: url(http://www.htmlhelp.com/stripe.gif) }  
BODY { background: url( stripe.gif ) }  
BODY { background: url("stripe.gif") }  
BODY { background: url(\"Ulalume\".png) } /* quotes in URL escaped */ /25/
```

5. PROBLEMS IN CSS AND INTERNET EXPLORER

Problems in Cascading Style Sheets are discussed in this chapter. These problems range from simple spelling errors in the specifications, to more complex questions as to whether CSS fulfils its intended role. The chapter is loosely organized along an axis of complexity; the first part describes how simple errors have been handled, and the rest discusses real and perceived problems in CSS.

5.1 Ghost Text bug

Internet Explorer 6 has a puzzling bug involving multiple floated elements; text characters from the last of the floated elements are sometimes duplicated below the last float. This bug is a real head banger because there seems to be nothing triggering it. However, by now everyone should know that IE needs no excuse to misbehave. /26/

The direct cause is nothing more than ordinary HTML comments, such as, `<!-- end left column -->`, sandwiched between floats that come in sequence. Apparently, the comments are hard for IE to digest when they occupy those positions, resulting in a kind of "screen diarrhea". HTML comments inside the floats do not cause the bug, nor do comments before or after the float series. Only comments residing between floats cause the bug. /26/

Solution:

1. Use `<!-- [IF! IE]>` tag surrounded comment
2. Remove comments
3. Add `style {display: inline;}` to floating;
4. Use negative margins on the floating div;
5. Add additional ` ` in the original text. /26/

5.2 Position Relative and Overflow Hidden

The key thing to notice in this CSS is the overflow set to the container, and the positioning set to element. Here's some code to demonstrate the problem:

```
<div id="container">
  <div id="a"></div>
  <div id="b"></div>
</div>
```

And the related CSS:

```
#container {
```

```
height:100px;
border:1px solid blue;
overflow: auto;
}
#a {
height:200px;
background-color:lightblue;
float:left;
width:60px;
}
#b {
position:relative;
height:200px;
background-color:pink;
width:60x;
}
```

Solution:

Add position: relative in parent element. /27/

5.3 Errors in the specifications

The CSS1 specification, which was first published as a W3C Recommendation in December 1996, was republished with all known errors corrected in January 1999. An appendix in the new document lists the changes and sorts them into three categories:

- Spelling and typographic mistakes. For example, a missing comma and right parenthesis were added. /28/
- Errors. For example, an invalid declaration (font-style: small-caps).
- Structure and organization. For example, the revised edition uses a new style sheet, and the appendix describing changes has been added.

A similar effort is planned for CSS2 but since it will also incorporate semantic changes (in addition to errors); it will most likely be given a new version number. /28/

5.4 Cascading problems

The cascading mechanism fulfils two important requirements for CSS. First, it allows both authors and users to influence the presentation of documents. Second, it provides fallback values when only partial style sheets are supplied, or when style sheets are missing. Still, the cascading mechanism has many associated problems. They are discussed in this section. Towards the end, some solutions are proposed. /28/

5.5 User interface problems

The user interface problem: Historically, browsers have provided users with very limited means of setting presentational preferences. Typical choices include the underlining and colouring of links. At a time when most of the presentation was hardcoded into the browser, this limited choice could be explained. With the arrival of CSS, however, most aspects of the presentation are configurable. Also, Microsoft's Internet Explorer for Windows does not provide a way of applying alternate style sheets. Most browsers do, however, provide a way to point to a file containing the user interface. /28/

5.6 Complexity problems

The site abundance problem: The web links thousands of servers and millions of pages. It is impossible to write one user style sheet that cascades well with all author style sheets on the web. This problem could have been addressed if browsers allowed site-specific user style sheets, but this has not been the case yet. /28/

The document debugger problem: Programmers use debuggers to find why a certain value is set at a certain point in a program. Similarly, a document debugger is needed to find out why a certain element/property combination has a certain style. This must be known in order to write a challenger rule to change the value. /28/

5.7 Problems in implementations

This thesis focuses on the design of CSS and other style sheet languages, and it is beyond its scope to analyze the level of CSS support in various browsers. Still, it must be mentioned that, from the point of view of web authors, the most pressing problem in the early years of CSS was the quality of CSS support in browsers. /28/

5.8 Bicubic Image Scaling

This is just a short link post, it seems IE has a trick to fix the image scaling. Normally if you forced an image to be smaller IE would screw it up but thanks to this nifty CSS trick this can be fixed.



Figure 2: Image Scaling in IE and Firefox

Solution:

```
img {-ms-interpolation-mode: bicubic ;} /29/
```


6. MOBILE CSS

Most web pages are written for desktop computers with large colour monitors. Mobile wireless devices typically have much smaller screens and presenting typical web pages on these units is a challenge. This chapter describes how CSS can be used to overcome the challenge. The solution is based on cascading: by enforcing a specially designed browser style sheet on all documents, the rendering of incoming documents is adjusted based on the constraints of the user's device. /30/

Mobile CSS standards are complicated, CSS Mobile Profile 1.0 and WAP CSS 1.0 both are subset of CSS 2.1, they have very similar content; the different is: WAP CSS 1.0 add some extensions on mobile devices.

Later, W3C make CSS Mobile Profile 1.0 and WAP CSS 1.0 integrated, absorbing some of the advantages of the WAP CSS1.0, introduced the CSS Mobile Profile 2.0 standard; it is also a subset of CSS 2.1. /30/

6.1 Usage and browser support

Most traditional phone browsers support handheld media types, including opera mini and windows mobile in IE. Mobile web site may directly through the handheld's media type to develop a mobile webpage CSS file: /31/

```
<link rel="stylesheet" media="handheld" href="">
```

There are several ways to make content fit on smaller screens. First, some browsers can zoom pages in and out. Zooming is a powerful way of getting the overview of complex web pages while also being able to magnify certain parts of the page. It is often used by visually impaired users to reach legible font sizes. Zooming out allows web pages written for desktop computers to be shown on small screens, but little content is legible when the page is zoomed out. The use of zooming typically requires the user to scroll extensively both horizontally and vertically.

Second, one can reformat content to better fit on small devices. Reformatting requires more processing of the content than does zooming; where zooming only changes the size of the elements on the screen while preserving the spatial relationships between the elements, reformatting means that the page is laid out in a new way that changes the spatial relationships between elements. /31/

6.2 CSS support in CSS Mobile Profile 2.0

Through this form, we can see support situation in CSS MP:

	Support	Not support
Selector	Type selectors, Child selectors Universal selector	:first-child、:hover、:lang() :first-letter、:first-line
Background And border	background, background- colour, background-image, background-repeat, background-attachment, background-position, border, border-width, border-colour, border-style	none
Position	position, top, right, bottom, left, z-index	None
List	list-style, list-style-image, list-style-type	list-style-position
Colour	colour	None
Font	font, font-family, font-style, font-variant, font-weight, font-size	None
Text	text-indent, text-align, text- decoration, text-transform, white-space	word-spacing, letter-spacing, unicode-bidi
User interface	utline, outline-colour, outline-style, outline-width	Cursor
Line	vertical-align	line-height

Table 1: Support situation in CSS MP. /31/

The strategy for reformatting content for small screens described in this chapter uses two aspects of CSS. First, cascading is used to enforce a browser style sheet over author and user style sheets. Second, CSS properties such as display, position, float and max-width are used to describe rendering in small-screen mode. The result is a browser that can display most web pages on a small screen. /31/

7. CONCLUSIONS

The topic of this thesis is style sheet languages for structured documents on the web. The hypothesis is that the web calls for different style sheet languages than does traditional electronic publishing. Further, the design of a style sheet language that fulfils the specific requirements of the web, namely Cascading Style Sheets, is described.

During I write the thesis, there still have some problems I can not solved. For example those problems I mentioned in chapter 5: Errors in the specifications, Cascading problems, User interface problems and Problems in implementations. I still tried to find solutions on internet and books. My thesis is all tell about the concept, it's very difficult to expound the really meaning of each theory by own word.

HTML was developed as a simple structured document format for the web. As web authors requested more presentational influence over their documents, HTML started developing into a presentational rather than a structural language. To stop this downwards slide on the ladder of abstraction, CSS was developed as a style sheet language for the web.

HTML has the right level of abstraction for a general-purpose mark up language: it is high enough on the ladder of abstraction to support presentation on a wide range of devices, and low enough for people to grasp easily the meaning of elements.

The CSS specification has three kinds of problems: errors, missing functionality and excessive functionality. The errors have been corrected by the editors who have published lists of errata and revised Recommendations. Also, test suites have been made available for implementers.

Most of the style sheet languages that were proposed for the web contained some innovative features. Some of these were later included in the development of CSS, but many were not. There are good reasons for not including all suggested features as the resulting specification will be too complex and may have conflicting functionality.

CSS is a style sheet language that has been designed for use on the web. It developed mainly from two of the early proposals for web style sheets, named CHSS and SSP. Some features from these early proposals were dropped in the course of developing CSS from proposal stage to W3C Recommendation stage.

Compared with other mature style sheet languages, CSS has some distinct and innovative features:

- Cascading allows several style sheets to influence the presentation of a document.
- CSS puts more emphasis on selectors than do other languages; rather than having expressions in the style sheet languages, the expressions are built into selectors.

Interoperability is one of the main goals for web specifications. Next to the specification itself, a comprehensive test suite is the most important tool for ensuring interoperability.

In retrospect, the introduction of CSS on the web would have been faster and easier if a test suite had been developed concurrently with the specification.

In this thesis accomplished time, regardless of my thesis is useful, I'm hard working on every paragraph, every line and every word. Even the four months of writing time is short, but I learned a lot of things.

8. REFERENCES

List of the sources:

- /1/ Sina, [WWW-document],
<<http://tech.sina.com.cn/introduction/html/homepage/C7-1.html>> 06.09.2010
- /2/ West 263, [WWW-document],
<<http://www.west263.com/info/html/wangzhanyunying/jianzhanjingyan/20090512/115132.html>> 06.09.2010
- /3/ W3C, [WWW-document],
<<http://www.w3.org/TR/REC-html40/present/styles.html>> 06.09.2010
- /4/ W3C, [WWW-document],
<www.w3.org/People/Berners-Lee/FAQ.html> 06.09.2010
- /5/ Issncn, [WWW-document],
<http://www.issncn.com/html/21291_1.htm> 10.09.2010
- /6/ 51CTO, [WWW-document],
<<http://developer.51cto.com/art/201008/222988.htm>> 10.09.2010
- /7/ FrameMaker 7.0 Tutorial, [WWW-document],
<http://www.io.com/~tcm/etwr2472/guides/frame/frame_structured.html> 18.09.2010
- /8/ Håkon Wium Lie, <<Cascading Style Sheets>> (PDF), p 44, 29 March, 2005
- /9/ Håkon Wium Lie, <<Cascading Style Sheets>> (PDF), p 46, 29 March, 2005
- /10/ Håkon Wium Lie, <<Cascading Style Sheets>> (PDF), p 60, 29 March, 2005
- /11/ Netease, [WWW-document],
<<http://tech.163.com/05/1020/11/20GKABHL00091589.html>> 27.09.2010
- /12/ Issncn, [WWW-document],
<http://www.issncn.com/html/21291_1.htm> 27.09.2010
- /13/ W3shcools, [WWW-document],
<http://www.w3schools.com/css/css_syntax.asp> 28.09.2010
- /14/ W3C, [WWW-document],
<<http://www.w3.org/TR/CSS2/selector.html>> 28.09.2010
- /15/ Use the tag selector, [WWW-document],
<<http://mail3server.net/help/files/%7BDE9DCC78-DE22-49F3-8368-8459303851BE%7D.htm>> 29.09.2010

- /16/ Web Design Group, [WWW-document],
<<http://www.htmlhelp.com/reference/css/font/font.html>> 29.09.2010
- /17/ Web Design Group, [WWW-document],
<<http://www.htmlhelp.com/reference/css/color-background/color.html>> 29.09.2010
- /18/ Web Design Group, [WWW-document],
<<http://www.htmlhelp.com/reference/css/color-background/background.html>>
29.09.2010
- /19/ Web Design Group, [WWW-document],
<<http://www.htmlhelp.com/reference/css/box/width.html>> 29.09.2010
- /20/ Web Design Group, [WWW-document],
<<http://www.htmlhelp.com/reference/css/box/height.html>> 02.10.2010
- /21/ Web Design Group, [WWW-document],
<<http://www.htmlhelp.com/reference/css/box/float.html>> 02.10.2010
- /22/ Web Design Group, [WWW-document],
<<http://www.htmlhelp.com/reference/css/box/clear.html>> 02.10.2010
- /23/ Web Design Group, [WWW-document],
<<http://www.htmlhelp.com/reference/css/classification/display.html>> 02.10.2010
- /24/ Web Design Group, [WWW-document],
<<http://www.htmlhelp.com/reference/css/units.html#urls>> 02.10.2010
- /25/ Web Design Group, [WWW-document],
<<http://www.htmlhelp.com/reference/css/units.html#urls>> 02.10.2010
- /26/ Explorer 6 Duplicate Characters Bug, [WWW-document],
<<http://www.positioniseverything.net/explorer/dup-characters.html>> 10.10.2010
- /27/ Web development, [WWW-document],
<http://snook.ca/archives/html_and_css/position_relative_overflow_ie/> 15.10.2010
- /28/ Håkon Wium Lie, <<Cascading Style Sheets>> (PDF), p 293, March 29, 2005
- /29/ Berea st, [WWW-document],
<http://www.456bereastreet.com/archive/200612/internet_explorer_and_the_css_box_model/> 18.10.2010
- /30/ Dustin Diaz, [WWW-document],
<<http://www.dustindiaz.com/min-height-fast-hack/>> 20.10.2010
- /31/ Webjx, [WWW-document],
<<http://www.webjx.com/css/divcss-24977.html>> 25.10.2010

9. LIST OF APPENDICES

Appendix 1: Inspiration

Appendix 2: Glossary

APPENDIX 1:

Style sheets constitute a wormhole into unspeakable universes.

–James D Mason, 1994

Style sheet languages are terribly under researched.

–Philip M Marden, Ethan V Munson, 1999

APPENDIX 2/1:

Binding

The process of combining a structured document with a style sheet with the intention of formatting the document into a final form presentation.

Cascading

The process of combining several style sheets and resolving conflicts between them.

Character

An entry in the Unicode Character Database.

Client

An application that communicates with a server over a network.

Content

The parts of a source document which are not markup. Also, the term refers to externally linked resources, for example, images and graphics.

Document

A collection of content, typically consisting of text, images and graphics. Traditionally, documents reach readers on printed paper, but electronic publishing is increasingly popular.

Element

The primary syntactic construct of a structured document.

Font

A typeface which can be classified by several characteristics, including family, size, weight and slant.

GUI

Graphical user interface.

Mark up

Tags and other symbols that, when embedded into content, form a source document.

Point

A unit of length equal to 1/72nd of an inch.

Property

A characteristic of an element which, when attached to a particular element and given a value, may influence the rendering the element.

Selector

A search pattern that identifies to what elements the corresponding declaration applies.

APPENDIX 2/2:

Structured document

A digital document consisting of hierarchical elements containing text and other content. The elements primarily represent the logical roles of the content rather than the presentation of the content.

Style sheet

In the context of electronic publishing, including this thesis, the following definition of a style sheet is offered:

A set of rules that associate stylistic properties and values with structural elements in a document, thereby expressing how to present the document. Style sheets generally do not contain content; are linkable from documents; and are reusable.

Style sheet language

A language that has syntax, selectors, properties, values and units, value propagation, and a formatting model. Style sheet languages are used to express style sheets.

Tag

A syntactic construct that marks the start and end of elements in HTML and other mark up languages.

Unit

A precisely specified quantity in terms of which values can be stated. Examples of units in style sheets are points and pixels.

URL

A web address.

Web browser

A computer program which fetches resources (for example text, graphics, and style sheets) from the web, decodes and assembles the resources, and presents the resulting content to a human user.

Web page

A document which is available on the web.

World Wide Web

A system of connected servers that uses HTTP to transfer documents and other information on request to browsers. The documents are typically written in HTML and include links to other documents.